Time Series

Austin Guimond

2022-11-22

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

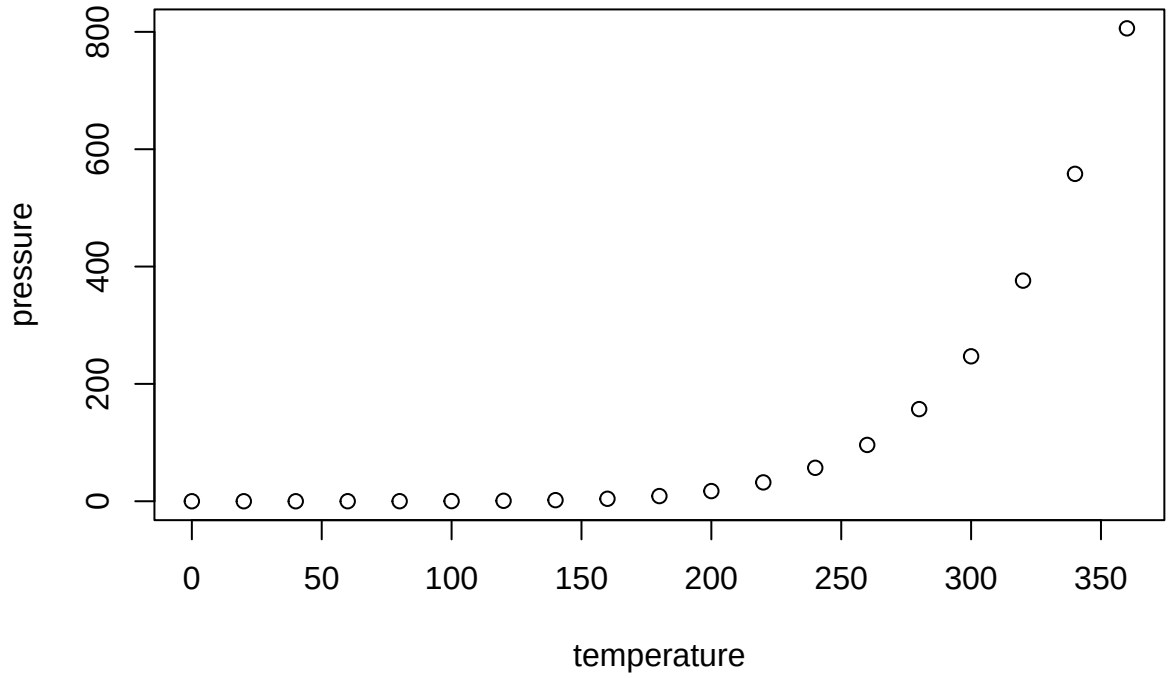
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
getwd()
```

```
## [1] "/home/guest/R/Patton_Guimond_ENV872_Final_Project"
```

```
#load packages
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(zoo)
```

```
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

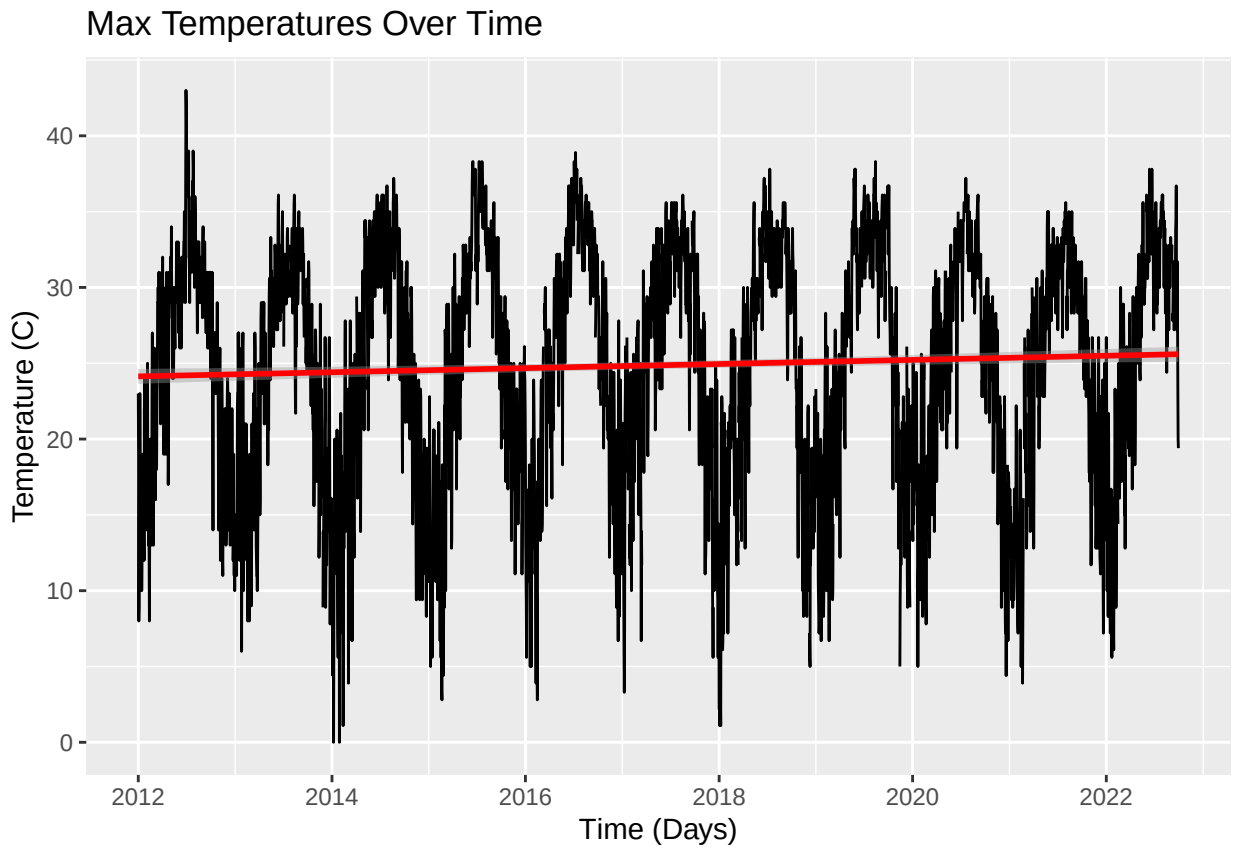
```
library(ggplot2)
library(trend)
library(Kendall)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(dplyr)

#Load Jackson Data
Jackson_Raw <- read.csv("~/R/Patton_Guimond_ENV872_Final_Project/Jackson_TenYear.csv",
                        stringsAsFactors = TRUE)
Jackson_Wrangle <- Jackson_Raw %>%
  select(YEAR, MO, DAY, TEMPC) %>%
  mutate('date' = make_date(year = YEAR, month = MO, day = DAY))

#Set as date
Jackson_Wrangle$date <- as.Date(Jackson_Wrangle$date, format = "%y/%m/%d")

#Group by date and find max daily temperature
Daily_High <- Jackson_Wrangle %>%
  group_by(date) %>%
  dplyr::summarize(value = max (TEMPC)) %>%
  as.data.frame()
#Plot max temperatures over time
MaxTemp_Plot <- ggplot(Daily_High, aes(x = date, y = value)) +
  geom_line()+
  geom_smooth(method=lm, col= 'red')+
 ggtitle("Max Temperatures Over Time")+
  xlab("Time (Days)") + ylab("Temperature (C)")
print(MaxTemp_Plot)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 74 rows containing non-finite values (stat_smooth).
```




```
#Look for NA values in data and remove
summary(Daily_High$value)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00   19.40   26.10   24.86   31.10   43.00      74
```

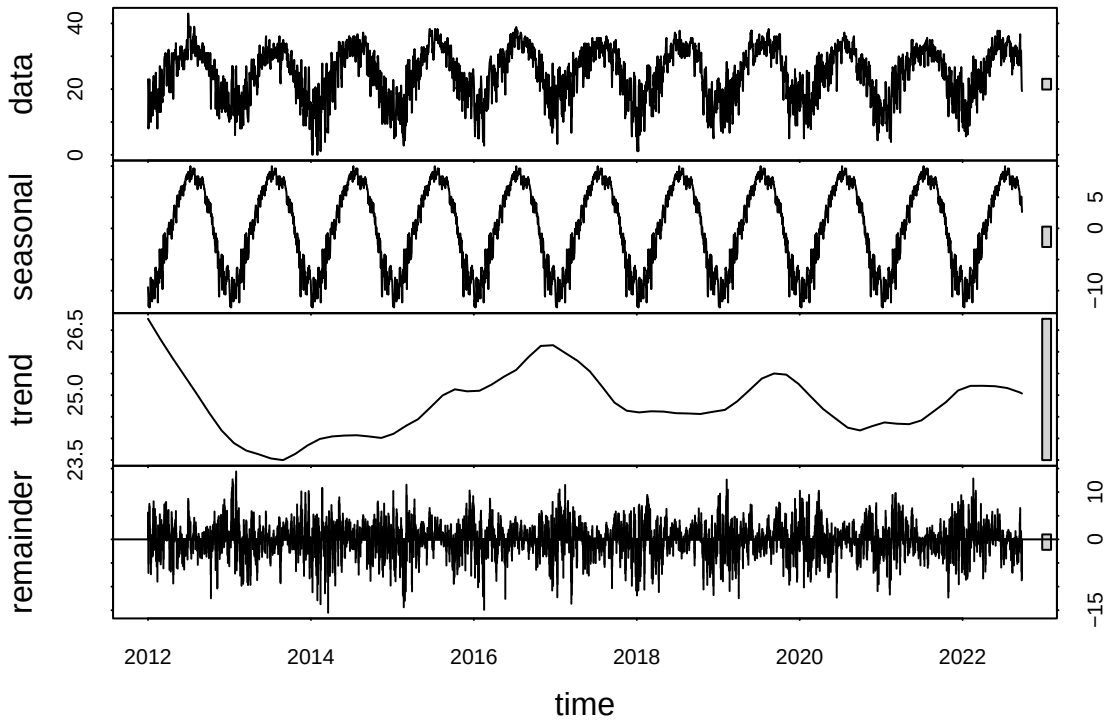
```
Clean_MaxTemp <-
  Daily_High %>%
  mutate(Temp_Clean = zoo::na.approx(value))
summary(Clean_MaxTemp)
```

```
##       date                value          Temp_Clean
##  Min.   :2012-01-01   Min.   : 0.00   Min.   : 0.00
##  1st Qu.:2014-09-06   1st Qu.:19.40   1st Qu.:19.40
##  Median :2017-05-13   Median :26.10   Median :26.10
##  Mean   :2017-05-14   Mean   :24.86   Mean   :24.87
##  3rd Qu.:2020-01-19   3rd Qu.:31.10   3rd Qu.:31.10
##  Max.   :2022-09-30   Max.   :43.00   Max.   :43.00
##                       NA's   :74
```

```
#Filter for date and NA omitted Temp
Max_Temp <- Clean_MaxTemp%>%
  select(date, Temp_Clean)
summary(Max_Temp)
```

```
##       date              Temp_Clean
##  Min.   :2012-01-01   Min.   : 0.00
##  1st Qu.:2014-09-06   1st Qu.:19.40
##  Median :2017-05-13   Median :26.10
##  Mean   :2017-05-14   Mean   :24.87
##  3rd Qu.:2020-01-19   3rd Qu.:31.10
##  Max.   :2022-09-30   Max.   :43.00
```

```
#Create Time series object and decompose

Daily_High_ts <- ts(Max_Temp$Temp_Clean, start = c(2012,01,01), frequency = 365)
Daily_High_decomp <- stl(Daily_High_ts,s.window = "periodic")
plot(Daily_High_decomp)
```




```
Daily_Temp_Trend <- Kendall::SeasonalMannKendall(Daily_High_ts)
summary(Daily_Temp_Trend)
```

```
## Score =  311 , Var(Score) = 55585
## denominator =  18761.71
## tau = 0.0166, 2-sided pvalue =0.18713
```

```
#Subtract seasonality and run seasonally adjusted Mann Kendall
Daily_Components <- as.data.frame(Daily_High_decomp$time.series[,1:3])
```

```
Daily_Components <- mutate(Daily_Components,
        Temp_C = Max_Temp$Temp_Clean,
        Date = Max_Temp$date)

TempSeasonAdj <- Daily_Components %>%
  mutate(Subtract.Season = Daily_Components$Temp_C - Daily_Components$seasonal)
summary(TempSeasonAdj)
```

```
##     seasonal             trend         remainder              Temp_C
##  Min.   :-12.69791   Min.   :23.50   Min.   :-15.538629   Min.   : 0.00
##  1st Qu.: -6.70109   1st Qu.:24.30   1st Qu.: -2.328569   1st Qu.:19.40
##  Median :  0.66055   Median :24.68   Median :  0.178168   Median :26.10
##  Mean   :  0.09144   Mean   :24.78   Mean   : -0.009183   Mean   :24.87
##  3rd Qu.:  6.68717   3rd Qu.:25.21   3rd Qu.:  2.472921   3rd Qu.:31.10
##  Max.   :  9.99058   Max.   :26.76   Max.   : 14.384892   Max.   :43.00
##       Date            Subtract.Season
##  Min.   :2012-01-01   Min.   : 8.489
##  1st Qu.:2014-09-06   1st Qu.:22.406
##  Median :2017-05-13   Median :24.939
##  Mean   :2017-05-14   Mean   :24.776
##  3rd Qu.:2020-01-19   3rd Qu.:27.316
##  Max.   :2022-09-30   Max.   :38.246
```

```
NonSeasonal_Temp_Trend <- Kendall::MannKendall(TempSeasonAdj$Subtract.Season)
summary(NonSeasonal_Temp_Trend)
```

```
## Score =  147891 , Var(Score) = 6680121344
## denominator =  7669158
## tau = 0.0193, 2-sided pvalue =0.070381
```

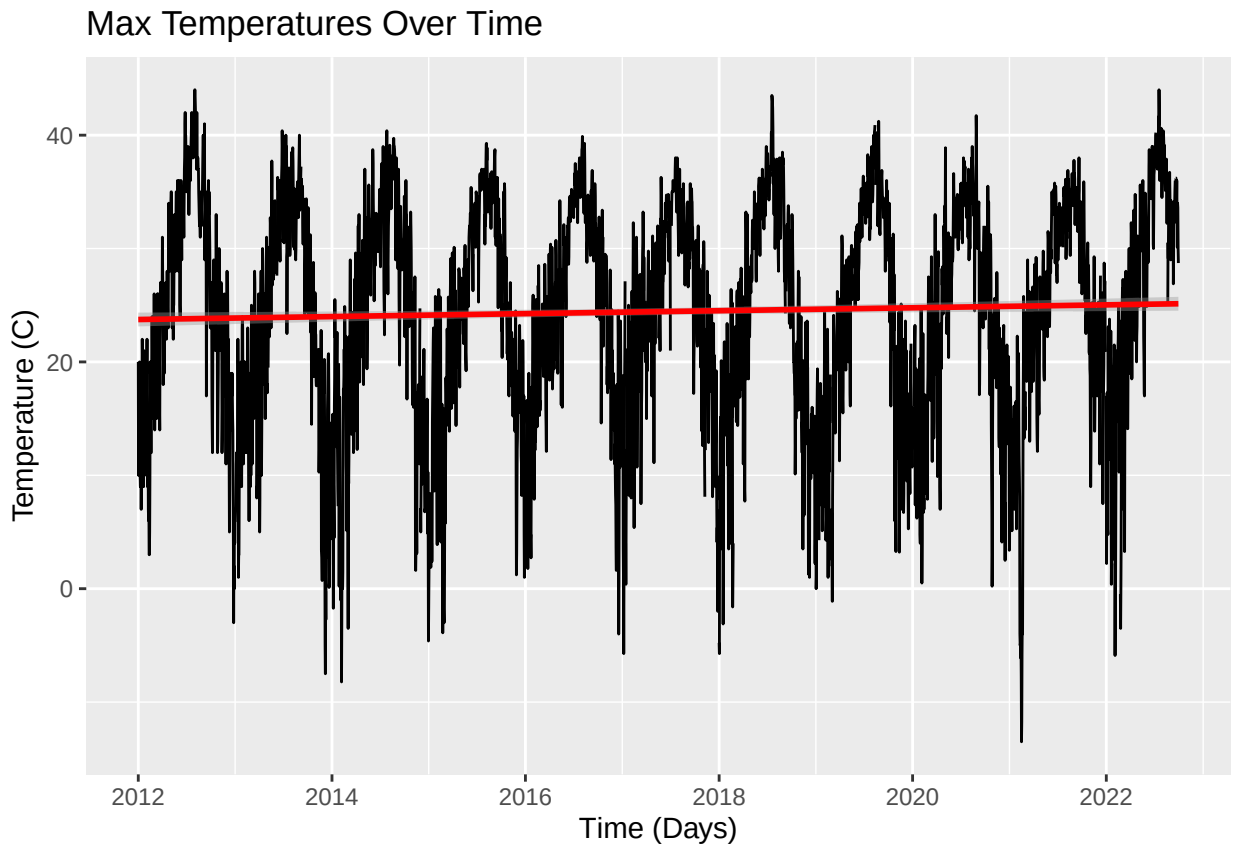
```
#Load Jackson Data
KFSI_Clean <- read.csv("~/R/Patton_Guimond_ENV872_Final_Project/KFSI_Clean.csv",
                       stringsAsFactors = TRUE)
KFSI_Wrangle <- KFSI_Clean %>%
  select(Year, Month, Day, Temperature..F.) %>%
  mutate('date' = make_date(year = Year, month = Month, day = Day)) %>%
           mutate('Temp_C' = ((Temperature..F.-32)/1.8 ))

#Set as date
KFSI_Wrangle$date <- as.Date(KFSI_Wrangle$date, format = "%y/%m/%d")
KFSI_Wrangle_Update <- KFSI_Wrangle %>%
  select(date, Temp_C)

#Group by date and find max daily temperature
Daily_High_KFSI <- KFSI_Wrangle_Update %>%
  group_by(date) %>%
  dplyr::summarize(value = max (Temp_C)) %>%
  as.data.frame()
#Plot max temperatures over time
KFSI_MaxTemp_Plot <- ggplot(Daily_High_KFSI, aes(x = date, y = value)) +
  geom_line()+
  geom_smooth(method=lm, col= 'red')+
```

```
 ggtitle("Max Temperatures Over Time")+
  xlab("Time (Days)") + ylab("Temperature (C)")
print(KFSI_MaxTemp_Plot)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 43 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
#Look for NA values in data and remove
summary(Daily_High_KFSI$value)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -13.50   17.78   25.72   24.43   32.39   44.00      43
```

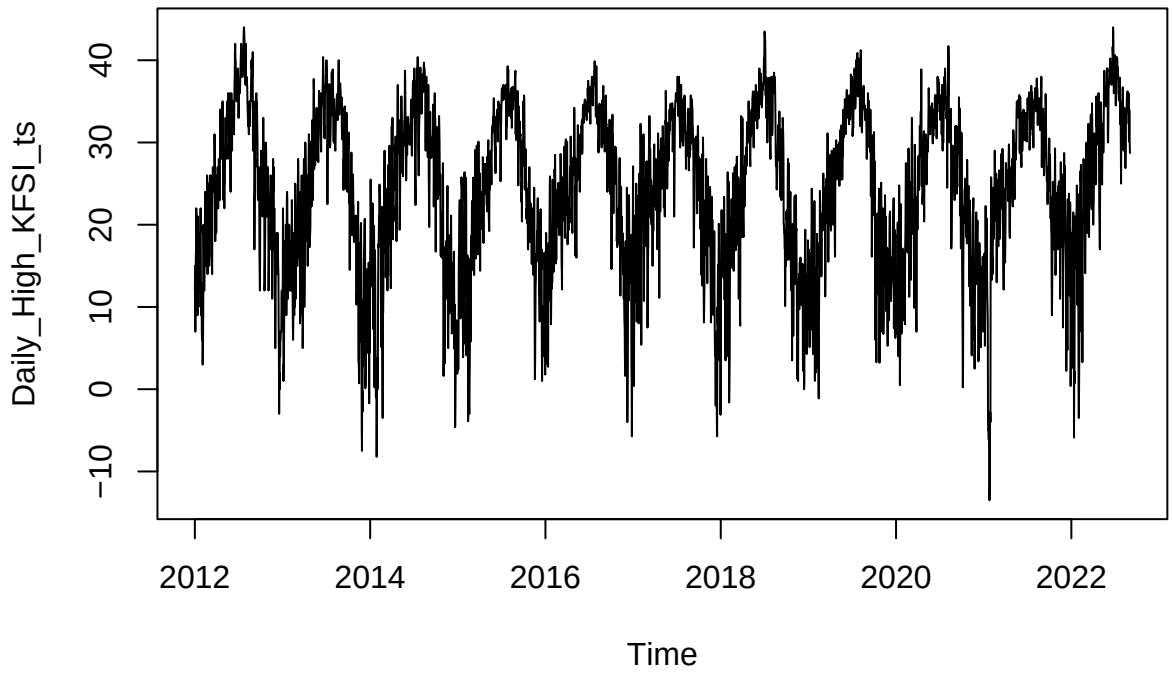
```
KFSI_Filtered_Date <- Daily_High_KFSI %>%
  filter(between(date, as.Date("2012-01-10"), as.Date("2022-09-30")))

Clean_MaxTemp_KSFI <-
  KFSI_Filtered_Date %>%
  mutate(Temp_Clean = zoo::na.approx(value))
summary(Clean_MaxTemp_KSFI)
```

```
##       date                value          Temp_Clean
##  Min.   :2012-01-10   Min.   :-13.50   Min.   :-13.50
##  1st Qu.:2014-09-10   1st Qu.: 17.89   1st Qu.: 17.78
##  Median :2017-05-13   Median : 25.72   Median : 25.72
##  Mean   :2017-05-17   Mean   : 24.45   Mean   : 24.44
##  3rd Qu.:2020-01-20   3rd Qu.: 32.39   3rd Qu.: 32.39
##  Max.   :2022-09-30   Max.   : 44.00   Max.   : 44.00
##                       NA's   :42
```

```
Clean_MaxTemp_KSFI <- Clean_MaxTemp_KSFI %>%
  select(date,Temp_Clean)
summary(Clean_MaxTemp_KSFI)
```

```
##       date              Temp_Clean
##  Min.   :2012-01-10   Min.   :-13.50
##  1st Qu.:2014-09-10   1st Qu.: 17.78
##  Median :2017-05-13   Median : 25.72
##  Mean   :2017-05-17   Mean   : 24.44
##  3rd Qu.:2020-01-20   3rd Qu.: 32.39
##  Max.   :2022-09-30   Max.   : 44.00
```

```
#Create Time series object and decompose

Daily_High_KFSI_ts <- ts(Clean_MaxTemp_KSFI$Temp_Clean, start = c(2012,01,01), frequency = 365)
Daily_High_KFSI_decomp <- stl(Daily_High_KFSI_ts,s.window = "periodic")
plot(Daily_High_KFSI_ts)
```



```
Daily_Temp_Trend_KFSI <- Kendall::SeasonalMannKendall(Daily_High_KFSI_ts)
summary(Daily_Temp_Trend_KFSI)
```

```
## Score =  259 , Var(Score) = 55246.33
## denominator =  18787.33
## tau = 0.0138, 2-sided pvalue =0.2705
```

```
#Subtract seasonality and run seasonally adjusted Mann Kendall
Daily_Components_KFSI <- as.data.frame(Daily_High_KFSI_decomp$time.series[,1:3])

Daily_Components_KFSI <- mutate(Daily_Components_KFSI,
        Temp_C = Clean_MaxTemp_KSFI$Temp_Clean,
        Date = Clean_MaxTemp_KSFI$date)

TempSeasonAdj_KFSI <- Daily_Components_KFSI %>%
  mutate(Subtract.Season = Daily_Components_KFSI$Temp_C - Daily_Components_KFSI$seasonal)
summary(TempSeasonAdj_KFSI)
```

```
##     seasonal            trend         remainder              Temp_C
##  Min.   :-15.4988   Min.   :22.91   Min.   :-23.595564   Min.   :-13.50
##  1st Qu.: -7.4985   1st Qu.:23.58   1st Qu.: -2.666245   1st Qu.: 17.78
##  Median :  0.4756   Median :24.14   Median :  0.423844   Median : 25.72
##  Mean   :  0.1504   Mean   :24.29   Mean   :  0.001064   Mean   : 24.44
##  3rd Qu.:  8.0260   3rd Qu.:24.80   3rd Qu.:  3.065015   3rd Qu.: 32.39
##  Max.   : 12.6716   Max.   :26.54   Max.   : 16.728707   Max.   : 44.00
##       Date            Subtract.Season
##  Min.   :2012-01-10   Min.   :-0.413
##  1st Qu.:2014-09-10   1st Qu.:21.575
##  Median :2017-05-13   Median :24.742
##  Mean   :2017-05-17   Mean   :24.291
##  3rd Qu.:2020-01-20   3rd Qu.:27.463
##  Max.   :2022-09-30   Max.   :41.731
```

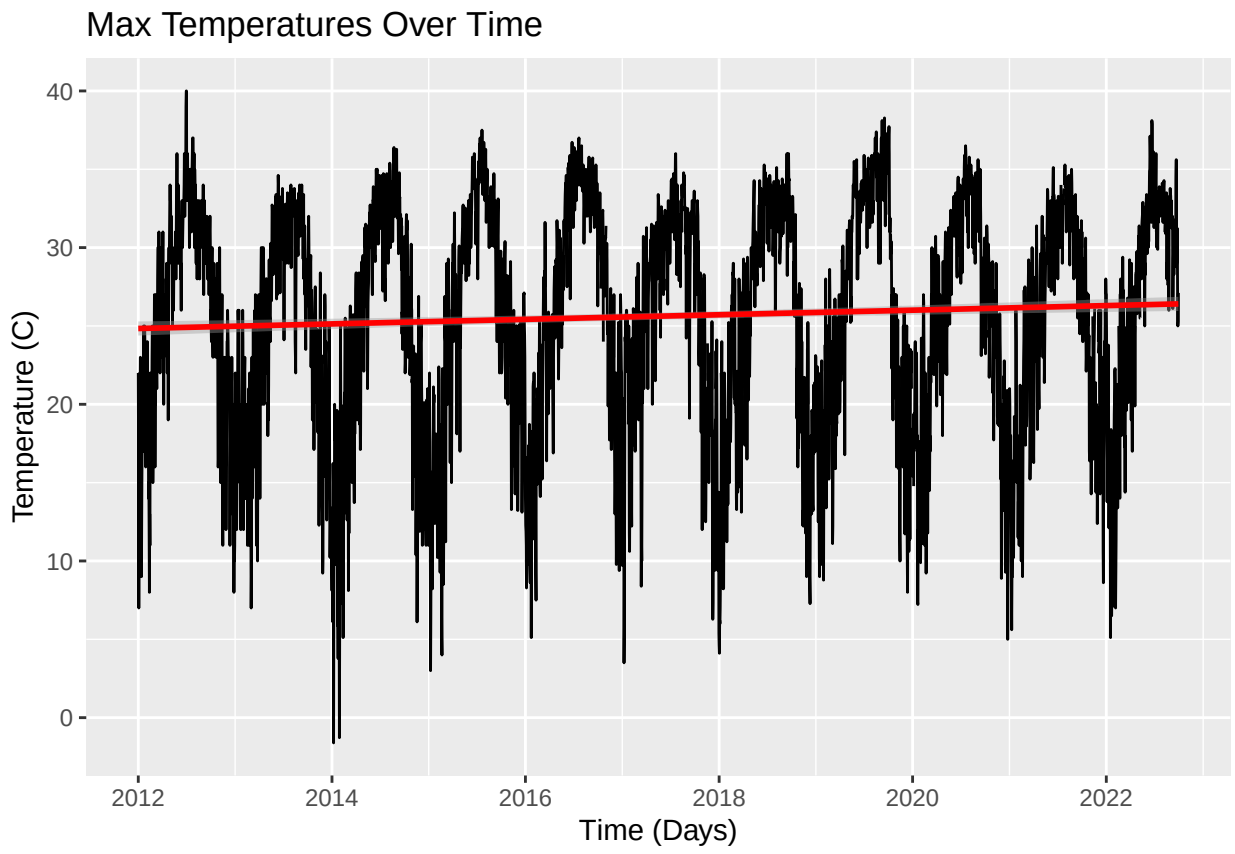
```
NonSeasonal_Temp_Trend_KFSI <- Kendall::MannKendall(TempSeasonAdj_KFSI$Subtract.Season)
summary(NonSeasonal_Temp_Trend_KFSI)
```

```
## Score =  99744 , Var(Score) = 6568209920
## denominator =  7583478
## tau = 0.0132, 2-sided pvalue =0.21843
```

```
#Load Data
KLSF_Clean <- read.csv("~/R/Patton_Guimond_ENV872_Final_Project/KLSF_Clean.csv",
                       stringsAsFactors = TRUE)
KLSF_Wrangle <- KLSF_Clean %>%
  select(Year, Month, Day, Temperature..F.) %>%
  mutate('date' = make_date(year = Year, month = Month, day = Day)) %>%
           mutate('Temp_C' = ((Temperature..F.-32)/1.8 ))

#Set as date
KLSF_Wrangle$date <- as.Date(KLSF_Wrangle$date, format = "%y/%m/%d")
KLSF_Wrangle_Update <- KLSF_Wrangle %>%
```

```
  select(date, Temp_C)

#Group by date and find max daily temperature
Daily_High_KLSF <- KLSF_Wrangle_Update %>%
  group_by(date) %>%
  dplyr::summarize(value = max (Temp_C)) %>%
  as.data.frame()
#Plot max temperatures over time
KLSF_MaxTemp_Plot <- ggplot(Daily_High_KLSF, aes(x = date, y = value)) +
  geom_line()+
  geom_smooth(method=lm, col= 'red')+
 ggtitle("Max Temperatures Over Time")+
  xlab("Time (Days)") + ylab("Temperature (C)")
print(KLSF_MaxTemp_Plot)
```

```
## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 28 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
#Look for NA values in data and remove
summary(Daily_High_KLSF$value)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -1.611  20.389  27.000  25.620  31.611  40.000      28
```

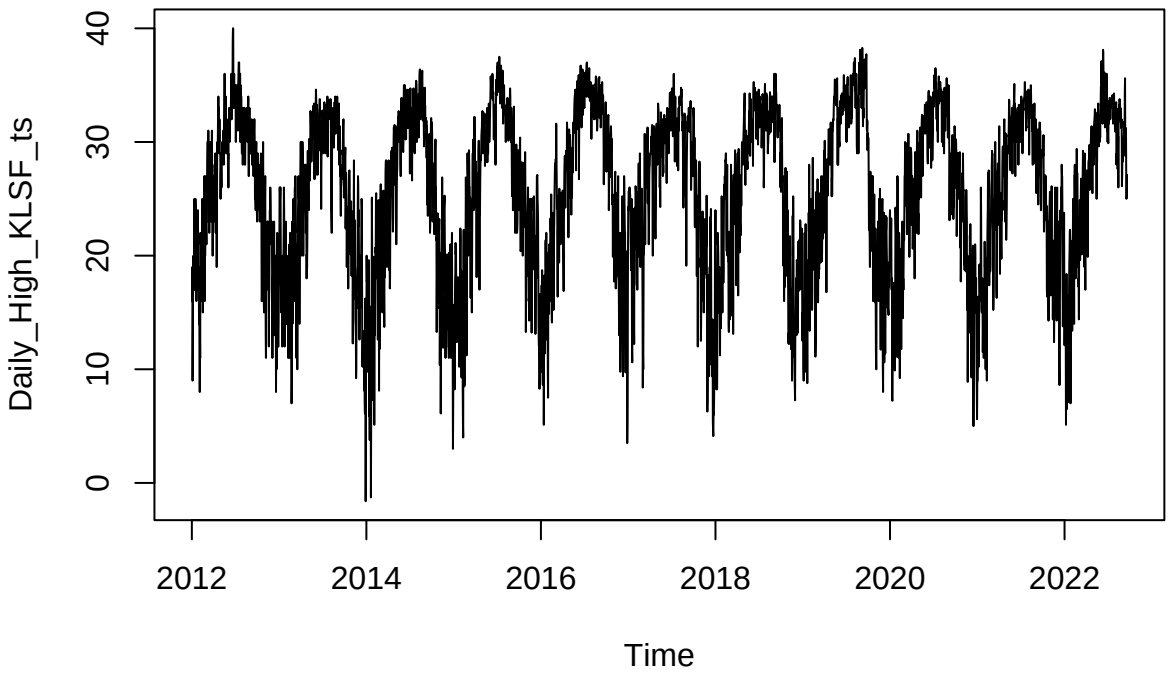
```
KLSF_Filtered_Date <- Daily_High_KLSF %>%
  filter(between(date, as.Date("2012-01-10"), as.Date("2022-09-30")))

Clean_MaxTemp_KLSF <-
  KLSF_Filtered_Date %>%
  mutate(Temp_Clean = zoo::na.approx(value))
summary(Clean_MaxTemp_KLSF)
```

```
##       date                value          Temp_Clean
##  Min.   :2012-01-10   Min.   :-1.611   Min.   :-1.611
##  1st Qu.:2014-09-15   1st Qu.:20.500   1st Qu.:20.389
##  Median :2017-05-21   Median :27.000   Median :27.000
##  Mean   :2017-05-20   Mean   :25.637   Mean   :25.624
##  3rd Qu.:2020-01-24   3rd Qu.:31.611   3rd Qu.:31.611
##  Max.   :2022-09-30   Max.   :40.000   Max.   :40.000
##                       NA's   :27
```

```
Clean_MaxTemp_KLSF <- Clean_MaxTemp_KLSF %>%
  select(date,Temp_Clean)
summary(Clean_MaxTemp_KLSF)
```

```
##       date              Temp_Clean
##  Min.   :2012-01-10   Min.   :-1.611
##  1st Qu.:2014-09-15   1st Qu.:20.389
##  Median :2017-05-21   Median :27.000
##  Mean   :2017-05-20   Mean   :25.624
##  3rd Qu.:2020-01-24   3rd Qu.:31.611
##  Max.   :2022-09-30   Max.   :40.000
```

```
#Create Time series object and decompose

Daily_High_KLSF_ts <- ts(Clean_MaxTemp_KLSF$Temp_Clean, start = c(2012,01,01), frequency = 365)
Daily_High_KLSF_decomp <- stl(Daily_High_KLSF_ts,s.window = "periodic")
plot(Daily_High_KLSF_ts)
```




```
Daily_Temp_Trend_KLSF <- Kendall::SeasonalMannKendall(Daily_High_KLSF_ts)
summary(Daily_Temp_Trend_KLSF)
```

```
## Score =  516 , Var(Score) = 55828
## denominator =  18911.23
## tau = 0.0273, 2-sided pvalue =0.028973
```

```
#Subtract seasonality and run seasonally adjusted Mann Kendall
Daily_Components_KLSF <- as.data.frame(Daily_High_KLSF_decomp$time.series[,1:3])

Daily_Components_KLSF <- mutate(Daily_Components_KLSF,
        Temp_C = Clean_MaxTemp_KLSF$Temp_Clean,
        Date = Clean_MaxTemp_KLSF$date)

TempSeasonAdj_KLSF <- Daily_Components_KLSF %>%
  mutate(Subtract.Season = Daily_Components_KLSF$Temp_C - Daily_Components_KLSF$seasonal)
summary(TempSeasonAdj_KLSF)
```

```
##     seasonal            trend         remainder              Temp_C
##  Min.   :-14.3996   Min.   :24.09   Min.   :-14.281380   Min.   :-1.611
##  1st Qu.: -5.6971   1st Qu.:25.12   1st Qu.: -1.994731   1st Qu.:20.389
##  Median :  0.4740   Median :25.47   Median :  0.242664   Median :27.000
##  Mean   :  0.1142   Mean   :25.52   Mean   : -0.005578   Mean   :25.624
##  3rd Qu.:  6.1801   3rd Qu.:25.90   3rd Qu.:  2.216205   3rd Qu.:31.611
##  Max.   :  8.7081   Max.   :27.32   Max.   : 11.897186   Max.   :40.000
```

```
##       Date            Subtract.Season
##  Min.   :2012-01-10   Min.   : 9.977
##  1st Qu.:2014-09-15   1st Qu.:23.424
##  Median :2017-05-21   Median :25.760
##  Mean   :2017-05-20   Mean   :25.510
##  3rd Qu.:2020-01-24   3rd Qu.:27.755
##  Max.   :2022-09-30   Max.   :37.298
```

```
NonSeasonal_Temp_Trend_KLSF <- Kendall::MannKendall(TempSeasonAdj_KLSF$Subtract.Season)
summary(NonSeasonal_Temp_Trend_KLSF)
```

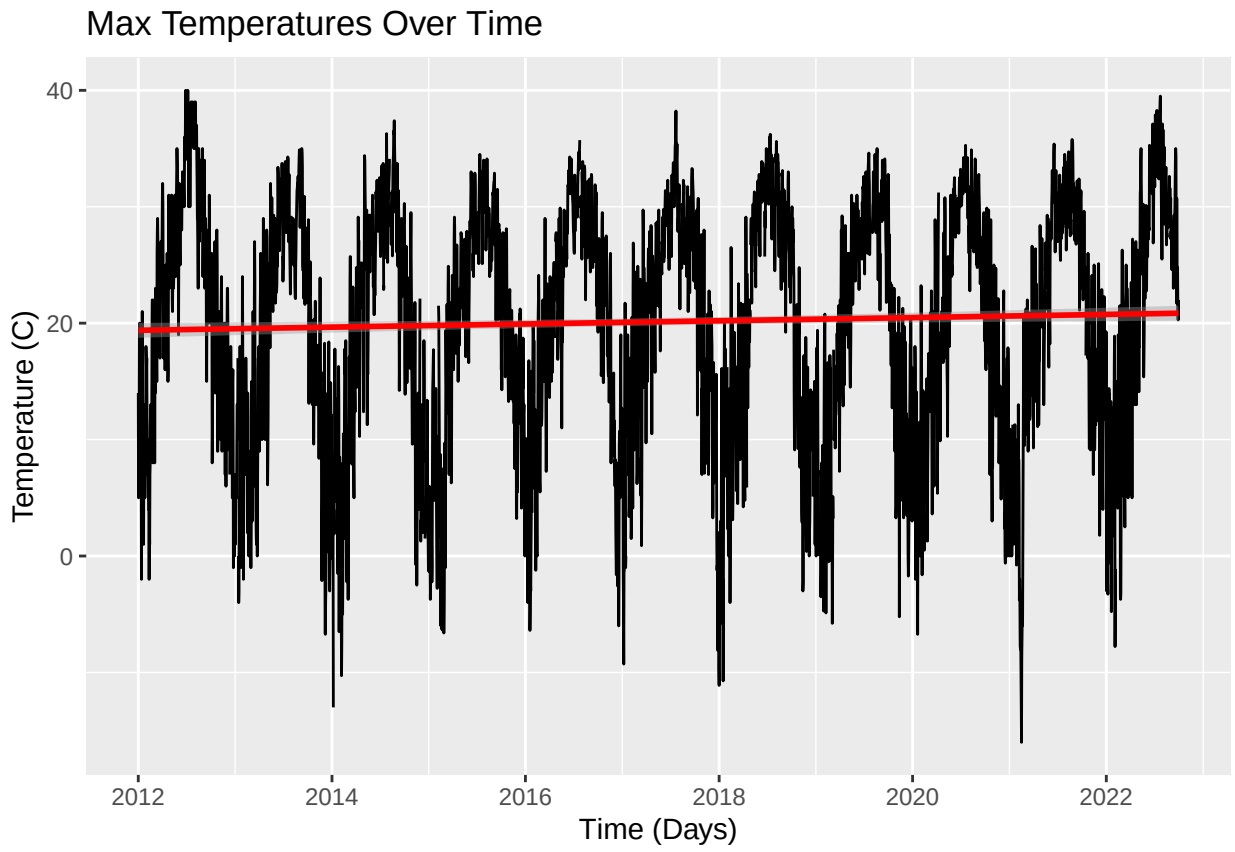
```
## Score =  213555 , Var(Score) = 6654576640
## denominator =  7649784
## tau = 0.0279, 2-sided pvalue =0.0088482
```

```
#Load Data
KTBN_Clean <- read.csv("~/R/Patton_Guimond_ENV872_Final_Project/KTBN_Clean.csv",
                       stringsAsFactors = TRUE)
KTBN_Wrangle <- KTBN_Clean %>%
  select(Year, Month, Day, Temperature..F.) %>%
  mutate('date' = make_date(year = Year, month = Month, day = Day)) %>%
           mutate('Temp_C' = ((Temperature..F.-32)/1.8 ))

#Set as date
KTBN_Wrangle$date <- as.Date(KTBN_Wrangle$date, format = "%y/%m/%d")
KTBN_Wrangle_Update <- KTBN_Wrangle %>%
  select(date, Temp_C)

#Group by date and find max daily temperature
Daily_High_KTBN <- KTBN_Wrangle_Update %>%
  group_by(date) %>%
  dplyr::summarize(value = max (Temp_C)) %>%
  as.data.frame()
#Plot max temperatures over time
KTBN_MaxTemp_Plot <- ggplot(Daily_High_KTBN, aes(x = date, y = value)) +
  geom_line()+
  geom_smooth(method=lm, col= 'red')+
 ggtitle("Max Temperatures Over Time")+
  xlab("Time (Days)") + ylab("Temperature (C)")
print(KTBN_MaxTemp_Plot)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 51 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
#Look for NA values in data and remove
summary(Daily_High_KTBN$value)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -16.00   13.00   21.78   20.12   28.39   40.00      51
```

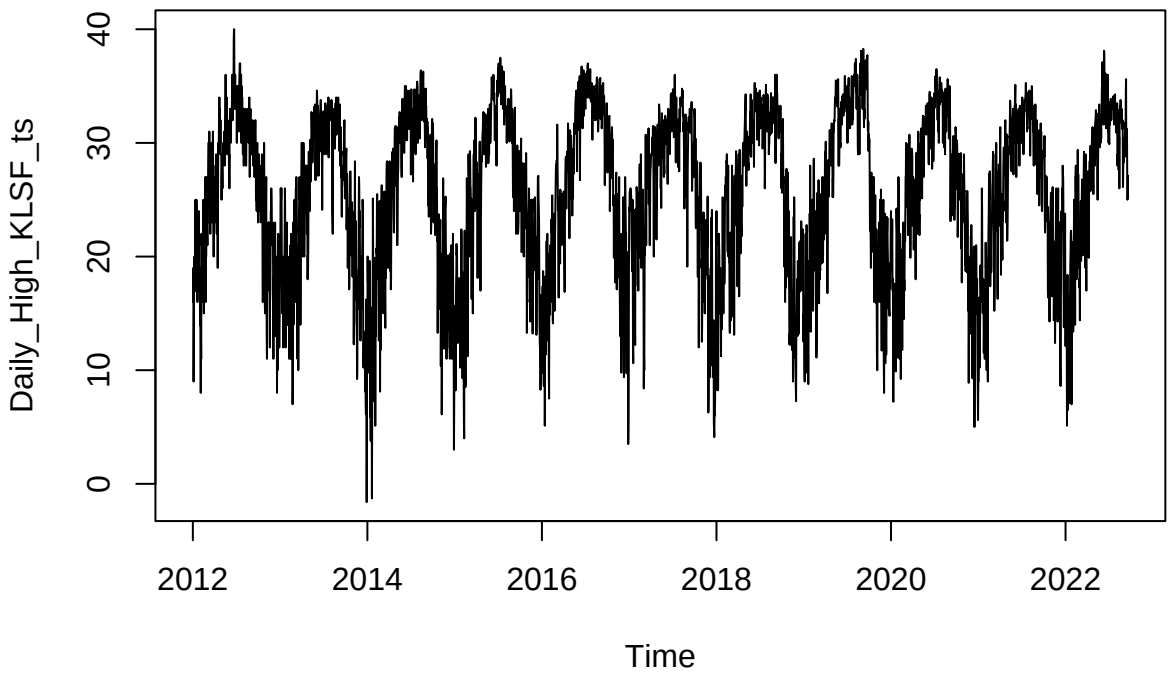
```
KTBN_Filtered_Date <- Daily_High_KTBN %>%
  filter(between(date, as.Date("2012-01-10"), as.Date("2022-09-30")))

Clean_MaxTemp_KTBN <-
  KTBN_Filtered_Date %>%
  mutate(Temp_Clean = zoo::na.approx(value))
summary(Clean_MaxTemp_KTBN)
```

```
##       date                value          Temp_Clean
##  Min.   :2012-01-10   Min.   :-16.00   Min.   :-16.00
##  1st Qu.:2014-09-16   1st Qu.: 13.00   1st Qu.: 13.00
##  Median :2017-05-21   Median : 21.89   Median : 21.89
##  Mean   :2017-05-21   Mean   : 20.14   Mean   : 20.10
##  3rd Qu.:2020-01-25   3rd Qu.: 28.39   3rd Qu.: 28.39
##  Max.   :2022-09-30   Max.   : 40.00   Max.   : 40.00
##                       NA's   :50
```

```
Clean_MaxTemp_KTBN <- Clean_MaxTemp_KTBN %>%
    select(date,Temp_Clean)
summary(Clean_MaxTemp_KTBN)
```

```
##       date              Temp_Clean
##  Min.   :2012-01-10   Min.   :-16.00
##  1st Qu.:2014-09-16   1st Qu.: 13.00
##  Median :2017-05-21   Median : 21.89
##  Mean   :2017-05-21   Mean   : 20.10
##  3rd Qu.:2020-01-25   3rd Qu.: 28.39
##  Max.   :2022-09-30   Max.   : 40.00
```

```
#Create Time series object and decompose

Daily_High_KLSF_ts <- ts(Clean_MaxTemp_KLSF$Temp_Clean, start = c(2012,01,01), frequency = 365)
Daily_High_KLSF_decomp <- stl(Daily_High_KLSF_ts,s.window = "periodic")
plot(Daily_High_KLSF_ts)
```




```
Daily_Temp_Trend_KLSF <- Kendall::SeasonalMannKendall(Daily_High_KLSF_ts)
summary(Daily_Temp_Trend_KLSF)
```

```
## Score =  516 , Var(Score) = 55828
## denominator =  18911.23
## tau = 0.0273, 2-sided pvalue =0.028973
```

```r
#Subtract seasonality and run seasonally adjusted Mann Kendall
Daily_Components_KLSF <- as.data.frame(Daily_High_KLSF_decomp$time.series[,1:3])

Daily_Components_KLSF <- mutate(Daily_Components_KLSF,
        Temp_C = Clean_MaxTemp_KLSF$Temp_Clean,
        Date = Clean_MaxTemp_KLSF$date)

TempSeasonAdj_KLSF <- Daily_Components_KLSF %>%
  mutate(Subtract.Season = Daily_Components_KLSF$Temp_C - Daily_Components_KLSF$seasonal)
summary(TempSeasonAdj_KLSF)
```

```
##     seasonal            trend         remainder              Temp_C
##  Min.   :-14.3996   Min.   :24.09   Min.   :-14.281380   Min.   :-1.611
##  1st Qu.: -5.6971   1st Qu.:25.12   1st Qu.: -1.994731   1st Qu.:20.389
##  Median :  0.4740   Median :25.47   Median :  0.242664   Median :27.000
##  Mean   :  0.1142   Mean   :25.52   Mean   : -0.005578   Mean   :25.624
##  3rd Qu.:  6.1801   3rd Qu.:25.90   3rd Qu.:  2.216205   3rd Qu.:31.611
##  Max.   :  8.7081   Max.   :27.32   Max.   : 11.897186   Max.   :40.000
##       Date            Subtract.Season
##  Min.   :2012-01-10   Min.   : 9.977
##  1st Qu.:2014-09-15   1st Qu.:23.424
##  Median :2017-05-21   Median :25.760
##  Mean   :2017-05-20   Mean   :25.510
##  3rd Qu.:2020-01-24   3rd Qu.:27.755
##  Max.   :2022-09-30   Max.   :37.298
```

```r
NonSeasonal_Temp_Trend_KLSF <- Kendall::MannKendall(TempSeasonAdj_KLSF$Subtract.Season)
summary(NonSeasonal_Temp_Trend_KLSF)
```

```
## Score =  213555 , Var(Score) = 6654576640
## denominator =  7649784
## tau = 0.0279, 2-sided pvalue =0.0088482
```